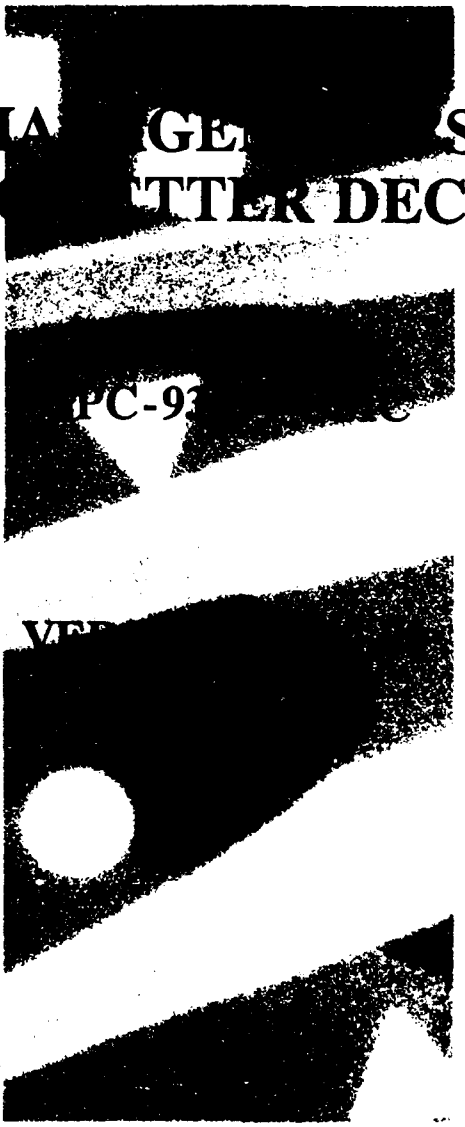


METRICS: MA GE S TOOL FOR MAKIN TTER DECISIONS

Software Productivity Consortium

METRICS: MANAGEMENT'S TOOL FOR MAKING BETTER DECISIONS

SPC-93034-CMC

VERSION 01.00.00

JUNE 1993

Robert Sulgrove

This material is based in part upon work sponsored by the Defense Advanced Research Projects Agency under Grant # MDA972-92-J-1018. The content does not necessarily reflect the position or the policy of the U.S. Government, and no official endorsement should be inferred.

This document accompanies a videotape of the same presentation recorded live at the Software Productivity Consortium in April 1993. It is recommended that the videotape be viewed with these viewgraphs at hand.

Produced by the
SOFTWARE PRODUCTIVITY CONSORTIUM

under contract to the
VIRGINIA CENTER OF EXCELLENCE
FOR SOFTWARE REUSE AND TECHNOLOGY TRANSFER
SPC Building
2214 Rock Hill Road
Herndon, Virginia 22070

METRICS:

MANAGEMENT'S TOOL FOR MAKING BETTER DECISIONS

ROBERT SULGROVE

APRIL 13, 1993

METRICS: MANAGEMENT'S TOOL FOR MAKING BETTER DECISIONS

About the presenter

Robert Sulgrove is a senior consulting analyst at NCR Corporation, where he is actively involved in applying software measurements to NCR's software development processes. Over the last several years, he has established a framework for software development which utilizes measurements during each development activity to improve the project team's decision making capabilities at that juncture. Bob has received several special recognation awards for the beneficial contributions that these efforts have had on software development at NCR.

Bob is active in various software metrics initiatives in the industry. He is Chairman of the IEEE working group that developed IEEE Std 1045-1992, IEEE Standard for Software Productivity Metrics. He is also a member of the IEEE Software Engineering Standards Subcommittee and the SEI Measurements Steering Committee. The foundation for the material presented at the SPC was obtained during his work with these industry initiatives.

Additionally, Bob has presented at IEEE, ASQC, QAI, and AIAA sponsored conferences, and has led and participated in several process improvement and standards panel discussions. Among his papers published are *Establishing a Practical Productivity Metric* and *Key Issues in Developing a Standard.*

Bob is currently on assignment at AT&T Bell Laboratories in Murray Hill, New Jersey, where he is continuing his work.

THE MANY VIEWS ABOUT METRICS

Observations from developing IEEE Std 1045-1992

IEEE Standard for Software Productivity Metrics

Measuring productivity vs. recognizing good productivity

Metrics definitions vs. applications

Macro metrics vs. primitive metrics

Personal favorites

SOFTWARE MEASUREMENT -- TYPICAL INITIAL ATTEMPTS

Measure only that which is easy to measure

- Delivered outputs
- Money spent

Predict whatever we wanted -- often without correlations

Linear relationships were assumed

Identical development processes were assumed

- Identical development methods and environments
- Constant mix of development tasks
- Constant proportion of documentation
- Constant proportion of tests

Early Popular Goals for Productivity Metrics

Comparing efficiency

- People
- Department
- Organization
- Company
- Industry

Displaying progress

- Project
- Improvement

Estimating new projects

IEEE 1045 conclusion was to define metrics, not to apply them

METRICS FOR ESTIMATING NEW PROJECTS

Simplicity vs. accuracy

- Single metric
- Perfect predictor -- perfect estimates require perfect information

Valid for all software projects

Estimate based on single composite variable plus adjustment factors

- Few adjustments
- Endless adjustments

Different mix of auxiliary activities

How accurate can an estimate be?

Macro estimating vs. micro estimating

THE SOFTWARE DEVELOPMENT PREDICAMENT

Software development is a very non-deterministic process

Many variabilities make development projects difficult

- Difficult to estimate and plan
- Difficult to keep on course
- Difficult to manage and control

Continual need to adjust the estimate and plan

- Effort to keep estimate and plan current is non-trivial
- Adjustments are needed most when everyone is busiest

DEVELOPMENT PROJECT VARIABILITIES

Product variability
- Unknowns
- Complexity

Requirements variability
- Changes
- Refinements

Project variability
- Unplanned additional activities
- Redo a completed activity
- Amount of rework required for completed activities
- Inter project dependencies
- Timely availability of acquired components

Personnel variability
- Expertise
- Experience
- Dedication

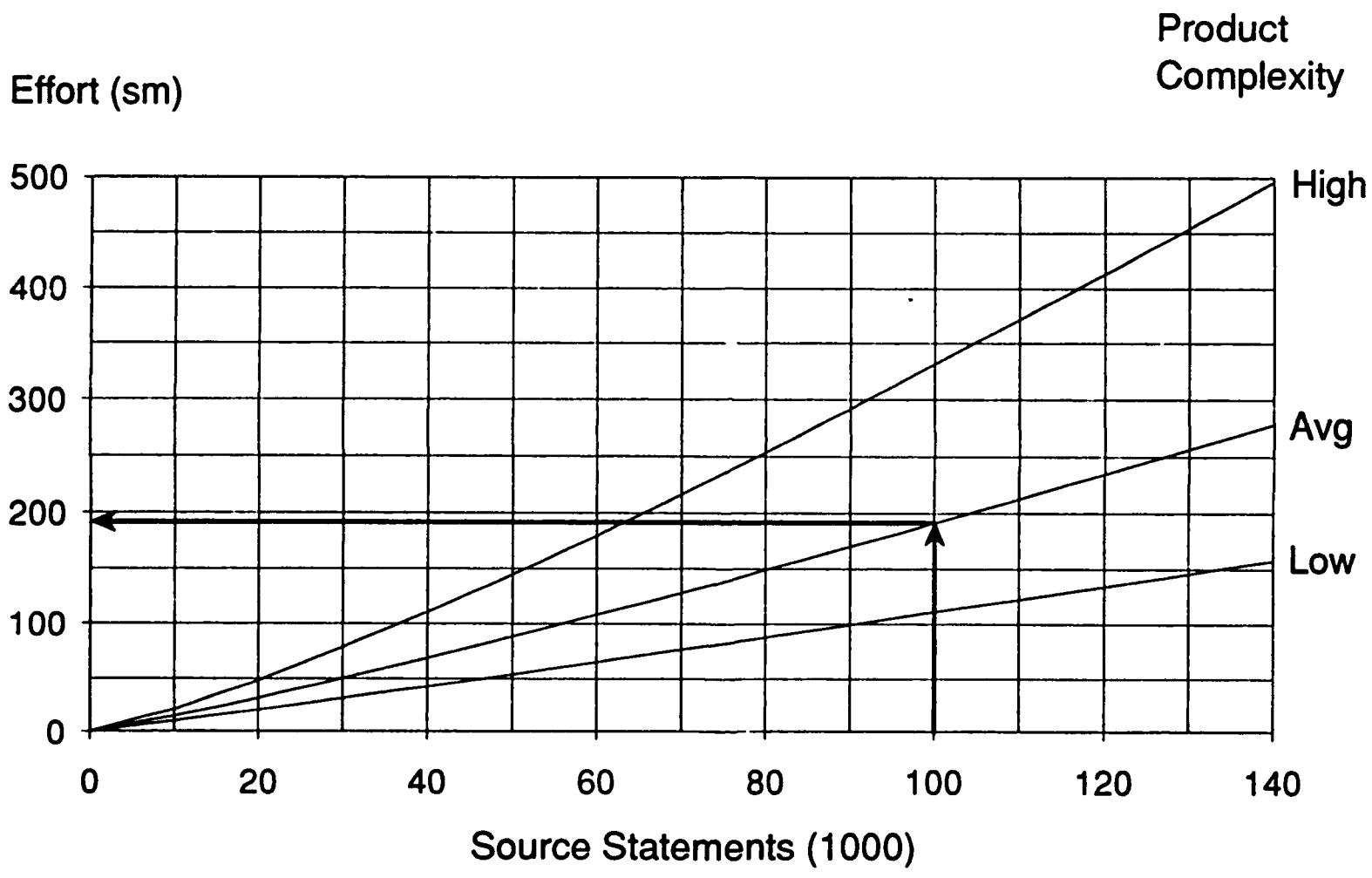
ESTIMATING THE PROJECT EFFORT - COCOMO

Product Complexity	Effort (staff months) SS = Source Statements / 1000
High	$Effort = 3.6 * (SS)^{1.20}$
Average	$Effort = 3.0 * (SS)^{1.12}$
Low	$Effort = 2.4 * (SS)^{1.05}$

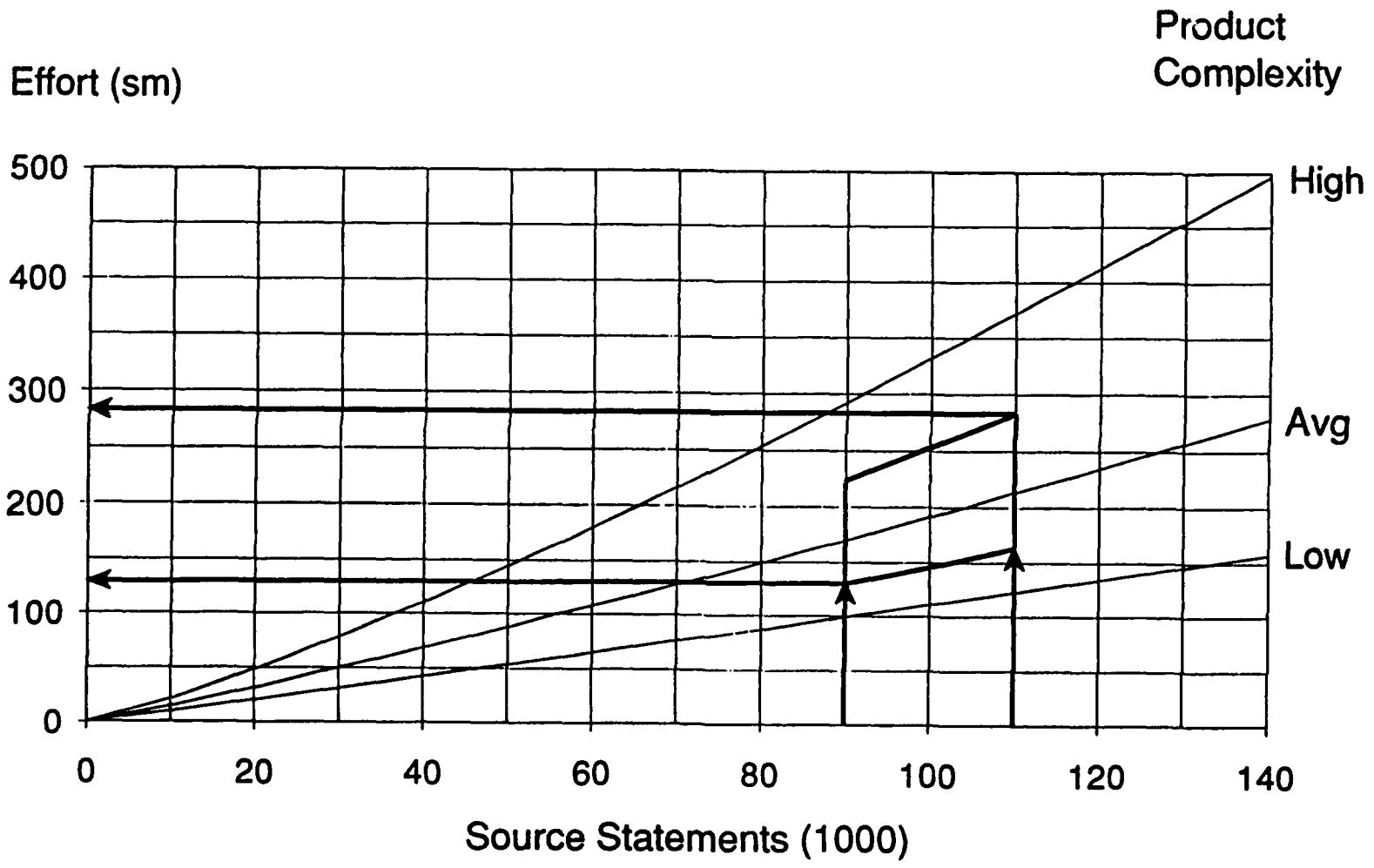
RECALIBRATED COCOMO ESTIMATING EQUATIONS

Product Complexity	Effort (staff months) SS = Source Statements / 1000
High	$Effort = 1.32 * (SS)^{1.20}$
Average	$Effort = 1.10 * (SS)^{1.12}$
Low	$Effort = 0.88 * (SS)^{1.05}$

ESTIMATING THE PROJECT EFFORT - COCOMO



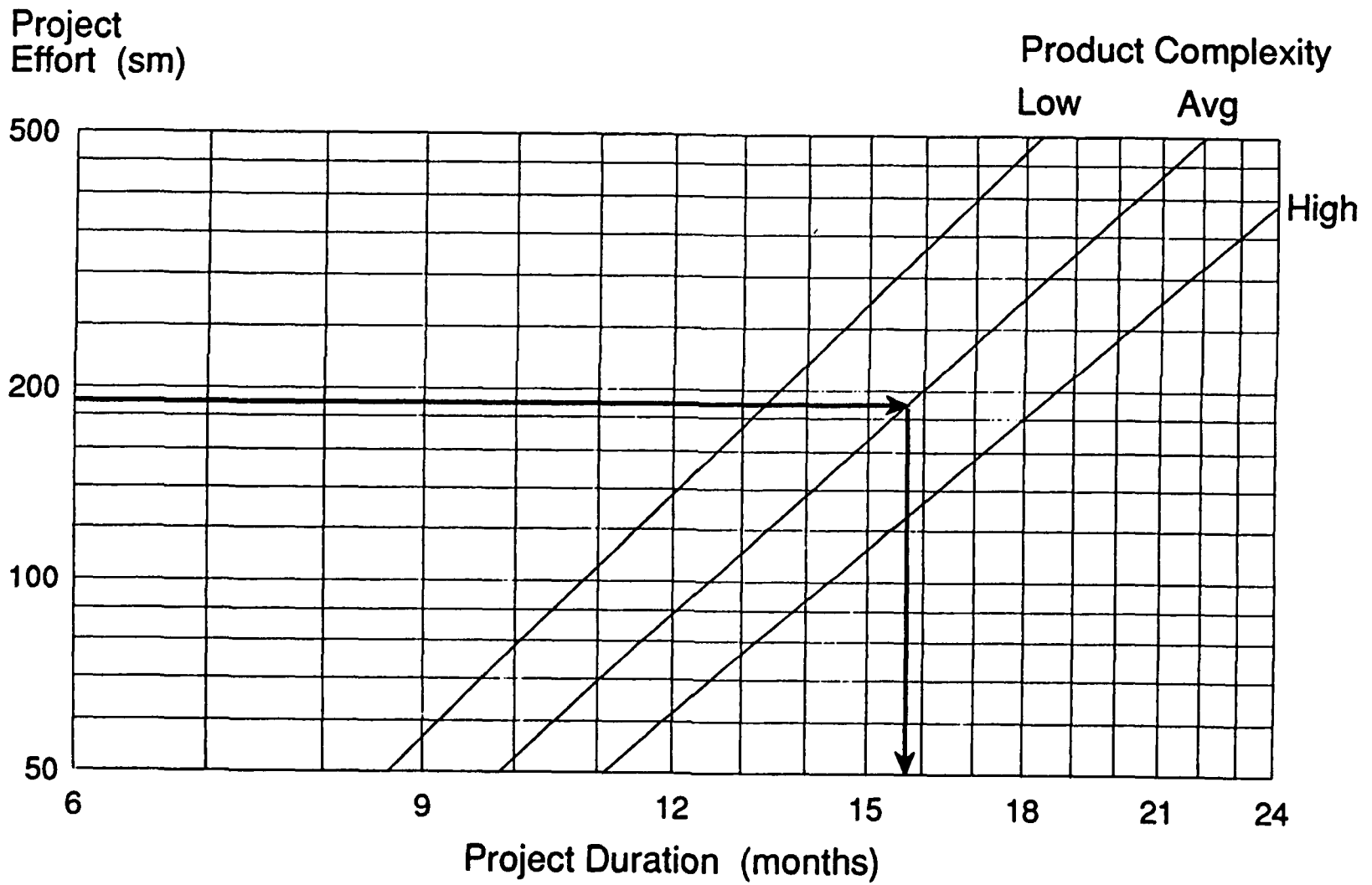
BOUNDING THE EFFORT ESTIMATE



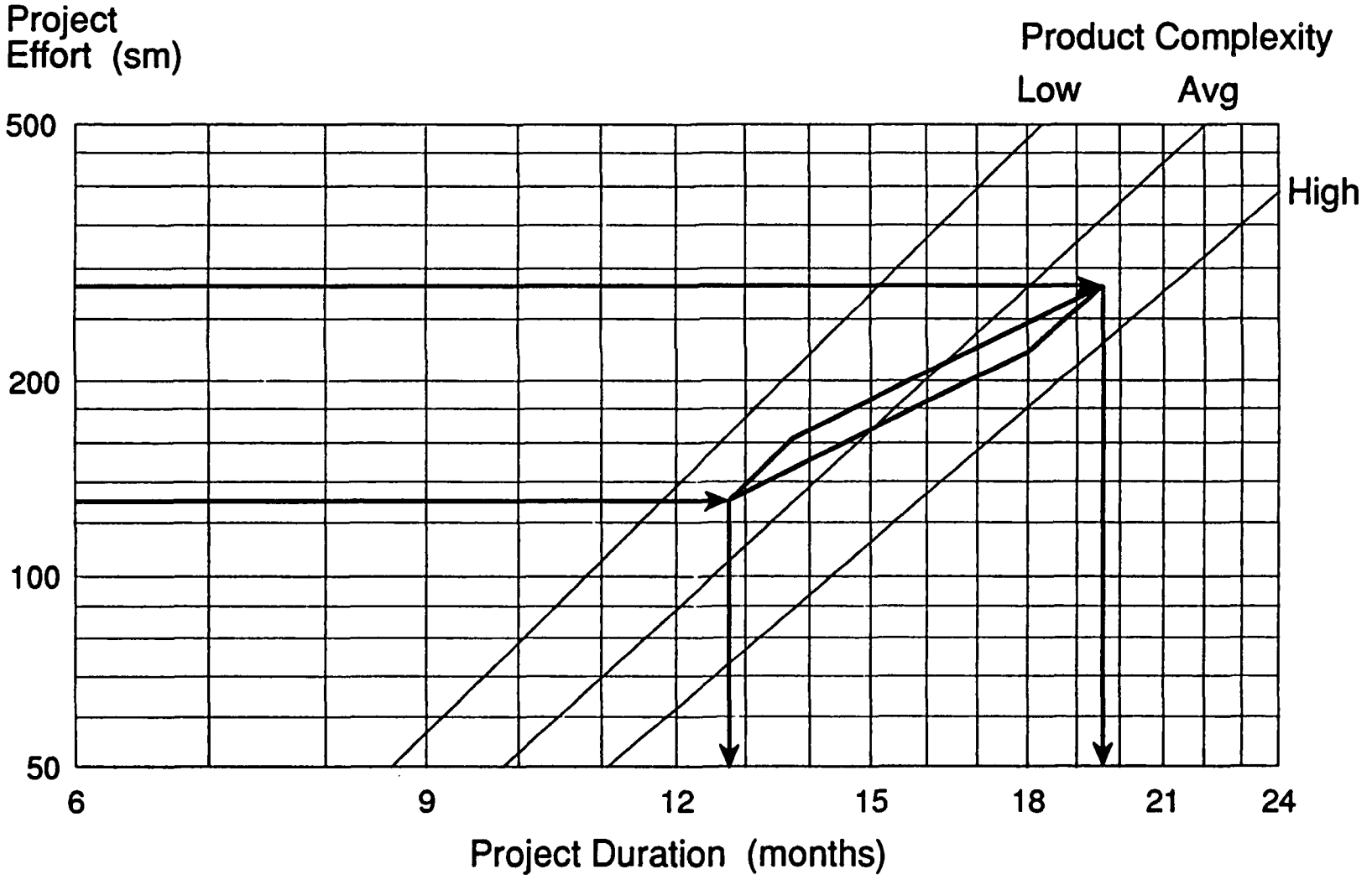
ESTIMATING THE PROJECT DURATION - COCOMO

Product Complexity	Duration (months) Effort (staff months)
High	$\text{Duration} = 2.5 * (E)^{0.38}$
Average	$\text{Duration} = 2.5 * (E)^{0.35}$
Low	$\text{Duration} = 2.5 * (E)^{0.32}$

ESTIMATING THE PROJECT DURATION - COCOMO



BOUNDING THE DURATION ESTIMATE



KEYS TO MANAGING PROJECT VARIABILITIES

Prepare solid, detailed plans

Closely monitor the project to quickly expose emerging problem situations

Analyze the available project data, and predict the project's future

Evaluate various project alterations to establish the best feasible plan adjustment

Continually refine the project plan based on new information

MEASUREMENT IS A POWERFUL MANAGEMENT TOOL

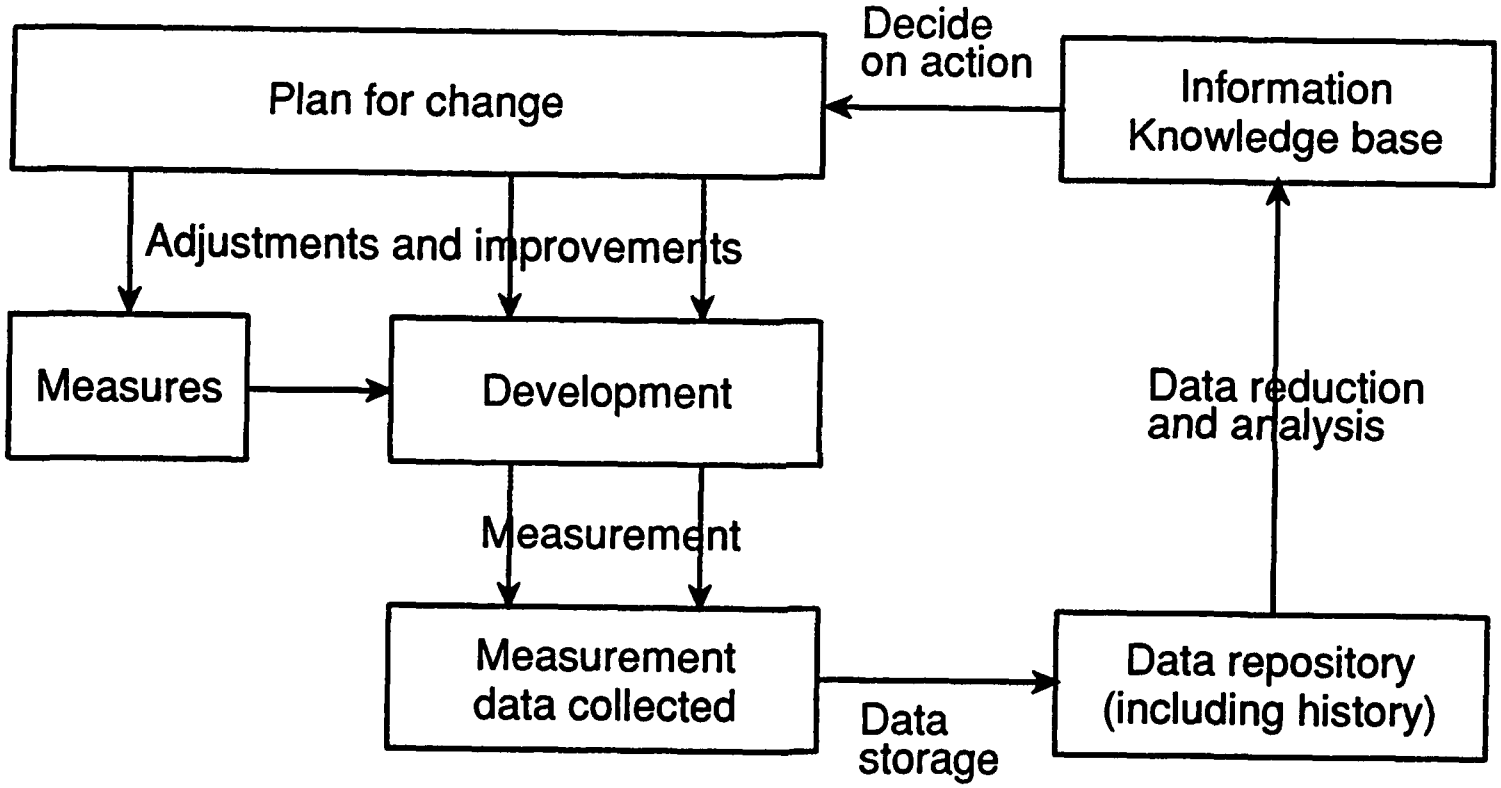
Metrics help to improve decision making capabilities

- New and better information is the basis for making better decisions
- Measurements augment judgement, but don't replace it
- Measurements alone are not enough, analysis is needed

Common uses

- Cost estimation
- Project control
- Quality control
- Process control
- Capability assessment

MEASUREMENT FEEDBACK LOOP



Measure → analyze → act → improve

PROJECT MANAGEMENT DECISION MAKING PROCESS

Search for potential problems

- Look for anything unexpected -- anything that deviates from the plan
- Early signs of a slowly emerging problem vs. random anomalous behavior
- Investigate to see if the potential problem is real

Determining the need for action

- Is the risk of continuing greater than the risk of changing?
- Will action(s) be more effective if taken now vs. later?

PROJECT MANAGEMENT DECISION MAKING PROCESS (CONT'D)

Identify possible changes to improve the situation

- Resources
- Schedule
- *Content*
- Quality

Study the impact of the proposed changes

- *Benefit of change vs. cost of implementing*
- Impact on other project activities and on other projects

Implement the changes

MEASUREMENT ISSUES

Data collection

- Large quantities of primitive data is needed
- Intrusion on people's time
- Accuracy and variability

Data reduction

- Converting the data into meaningful information
- Comparing to baselines

Data analysis

- Evaluating various possibilities
- Correlating cause-effect relationships

Deciding to take action

- Information and analysis must be convincing
- Predictors must be accepted

IMPLEMENTING SOFTWARE MEASUREMENTS

The vicious implementation circle

Define metrics -->

Propose uses -->

Try -->

Validate the use of metrics

Implementing a new measurement goal requires a long lead time

- Identify new metrics to be collected
- Begin data collection
- Wait until meaningful data has been obtained
- Analyze and interpret the data, and take action

IMPLEMENTING SOFTWARE MEASUREMENTS (CONT'D)

Define a family of metric primitives which can be used for most conceivable GQM measures

Begin collecting data

As new issues and goals emerge, some data will already be available

The primitive metrics are combined into useful measures which address specific issues toward satisfying the stated goals

Use SEI Capability Maturity Model to focus on topics

- Level 2 -- Basic project control
- Level 3 -- Process control
- Level 4 -- Product control

THE METRIC PRIMITIVES

Value -- revenue generated or benefit derived

Cost -- total cost of the software product

Functionality -- function points

Size -- source statements, document pages, design diagram nodes

Effort -- staff-hours to produce

Time -- interval to complete

Complexity characterization -- nominal or ordinal measures

Defects -- counts by criticality

RELATIONSHIP AMONG THE METRIC PRIMITIVES

$$\text{Value} = \underbrace{\frac{\text{excellence}}{\text{function}}}_{\text{Quality}} * \underbrace{\frac{\text{function}}{\text{size}}}_{\text{Effectiveness}} * \underbrace{\frac{\text{size}}{\text{effort}}}_{\text{Efficiency}}$$

Productivity

Questions or comments on content should be directed to:

Robert N. Sulgrove
NCR Corporation
Technology and Development Division
1700 South Patterson Boulevard
Dayton, Ohio 45479
(513) 445-1064

Or to:

John Gaffney
Software Productivity Consortium
2214 Rock Hill Road
Herndon, VA 22070
(703) 742-7116

Send feedback on the Consortium's Video Program and orders for video products to:

Technology Transfer Clearinghouse
Software Productivity Consortium
2214 Rock Hill Road
Herndon, VA 22070
(703) 742-7211